



Instituto Tecnológico de Costa Rica

Área Académica de Ingeniería en Computadores

Curso: CE-3101 Bases de Datos

Profesor: Marco Rivera Meneses

Plan de Trabajo

StraviaTEC

Estudiantes:

Gabriel Vargas López

Víctor Castrillo Muñoz

Yendry Badilla Gutiérrez

Mauricio Calderón Chavarría

Grupo: 1

Fecha: 6 de Junio del 2022

Índice

| | |
|--|----|
| Introducción..... | 3 |
| Análisis del Problema. | 4 |
| Modelo Conceptual (Planteo de solución)..... | 4 |
| Trabajo individual y en equipo. | 6 |
| Metas del proyecto | 6 |
| □ Aplicación Web Vista Administrador..... | 6 |
| □ Aplicación Web Vista Cliente | 6 |
| □ Aplicación Móvil..... | 6 |
| □ Servicio API | 6 |
| □ Base De Datos | 7 |
| Roles | 7 |
| Reglas | 8 |
| Cronogramas..... | 8 |
| Bitácora | 9 |
| Aprendizaje Continuo | 10 |
| Estrategia de aprendizaje..... | 10 |

Introducción

StraviaTEC es una aplicación pensada para deportistas que busca generar un entorno en donde sus clientes puedan almacenar sus registros de diferentes tipos de actividades que realizan, para esto se ha decidido realizar una aplicación Web que le permita a los deportistas almacenar dicha información, así como poder interactuar con otras personas de la plataforma, crear grupos interactivos, inscribirse en carreras y retos así como poder compartir los entrenamientos realizados con sus seguidores dentro de la misma plataforma.

También StraviaTEC ha pensado en la forma en como su usuario llevaran a cabo el registro de sus actividades y para ello ha decidido crear un apartado en la web para este fin, así como el desarrollo de una aplicación móvil para poder rastrear con ayuda del GPS todas las actividades realizadas por el deportista. Para ello a continuación se detalla como se plantea el desarrollo de la aplicación StraviaTEC.

Análisis del Problema.

Modelo Conceptual (Planteo de solución)

Para esta sección se decidió mostrar el análisis y elaboración de la solución a través del modelo conceptual, para esto el equipo decidió realizar únicamente un solo modelo conceptual, sin embargo, durante la elaboración de este se decidió recopilar las decisiones individuales con el fin de evidenciar la toma de decisiones grupal, en este anexo únicamente se decidió agregar tres problemas de análisis realizados durante la ejecución del modelo conceptual, la totalidad del análisis se mostrara en la documentación final del proyecto, los tres problemas elegidos se muestran a continuación

1. Decisión de la relación entre inscripción y reto.

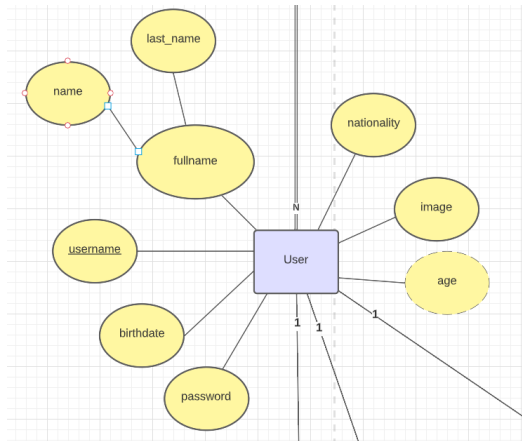
Inicialmente se creó la relación de inscripción con reto y carrera, sin embargo, se decidió eliminar la relación inscripción-reto ya que el reto es un evento gratuito, en cambio para participar en una carrera se debe realizar un pago.

2. Se eliminó entidad Deportista.

Durante el desarrollo del modelo conceptual, varios compañeros sugirieron realizar una entidad deportista, esto para poder separar al deportista del usuario que interactúa en la web y la aplicación, pero finalmente por decisión grupal esta idea fue descartada ya que siguiendo los detalles mencionados en la especificación, únicamente se restringe a que cada usuario ya registrado en la web/aplicación pueda inscribirse a una carrera o reto y esto no pueda hacerlo para un tercero, factor que anula la funcionalidad de la entidad y por esto se decide eliminar

3. Discusión por atributos compuestos (fecha y nombre completo)

Se tiene la necesidad de guardar el nombre a los usuarios, pero surgió la duda al ver si se guardaban los apellidos como atributo compuesto del nombre o ver si se guarda el nombre completo en un solo atributo.



Por el momento, decidimos guardar el nombre completo en un atributo compuesto, donde hay un atributo para nombre y otro atributo donde se guardan ambos apellidos.

Trabajo individual y en equipo.

Metas del proyecto

- ✓ **Aplicación Web Vista Administrador:** Crear una aplicación web que permite las funcionalidades de administrar los siguientes aspectos de la aplicación StraviaTEC, estos serían, gestionar carreras, aceptar inscripciones y gestionar retos y grupos, generar diferentes tipos de reporte como los participantes de carrera y posiciones finales. Además, esta aplicación tendrá la capacidad de interactuar con una base de datos a través de un servicio API.
- ✓ **Aplicación Web Vista Cliente:** Crear una aplicación web que permita a una comunidad de deportistas registrados mantener sus actividades registradas y compartir información sobre sus logros en cuanto a las actividades realizadas, accediendo a información de una base de datos. Además, esta aplicación tendrá la capacidad de interactuar con la base de datos a través de un servicio API.
- ✓ **Aplicación Móvil:** Crear una aplicación móvil con todos los atributos necesarios para que el cliente pueda realizar el registro de actividades deportivas. La aplicación debe mostrar el tiempo que lleva desde el inicio de una actividad y los kilómetros recorridos de la misma. También tendrá la capacidad de almacenar la ruta de GPS por donde transita el deportista. Los datos se almacenarán en SQL lite que permitirá guardar los datos localmente y sincronizarlos con la base de datos principal cuando exista conexión.
- ✓ **Servicio API:** Crear un servicio API para centralizar los datos, y las distintas funcionalidades requeridas ya sea por la aplicación web o la aplicación móvil que serán administrados por medio de métodos como Get, Post, Delete, etc. También debe proporcionar la funcionalidad para que la base de datos de la aplicación móvil pueda sincronizarse con la base de datos conectada al API. Además, la capa de servicio será implementada utilizando Azure o AWS, que se encarga de realizar las funciones principales del sistema.

- ✓ **Base De Datos:** Crear una base de datos capaz de almacenar todos los datos necesarios sobre las carreras, usuarios, inscripciones y gestión de grupos, entre otros, haciendo uso de SQL Server.

Roles

| Nombre | Roles |
|----------|---|
| Mauricio | App Web Vista Administrador y Vista Cliente Encargado de las reuniones |
| Yendry | App Movil con SQL Lite Encargado de la documentación |
| Víctor | REST API Y Base de Datos SQL Server Encargado del cumplimiento del cronograma Punto de contacto |
| Gabriel | Aplicación móvil |

Reglas

- Indicar si existe algún problema para la entrega de alguna tarea.
- No postergar la entrega de una tarea más de dos veces. De ser así, debe coordinar con el equipo para encontrar una manera de resolver el problema o realizar cambios de tarea.
- Durante las reuniones se debe mostrar evidencia de las tareas en desarrollo.
- Comunicar al equipo cualquier situación que pueda afectar el avance del proyecto.
- Al hacer merge a la branch principal, todos los miembros del grupo deben comunicarlo y verificar que el código sea funcional.
- Se debe hacer un commit donde se realiza la documentación interna después del commit del código realizado
- Durante una reunión, si alguno de los miembros del equipo se siente agotado o nota agotamiento laboral en uno de los miembros/as, este debe descansar el tiempo que considere necesario antes de continuar en la reunión.
- Respeto por todas y todos los miembros del equipo.

Cronogramas

A continuación, se presenta una lista de cronogramas según las secciones en que se dividió el proyecto

Bitácora

| Fecha de registro | Encargado de la tarea | Tipo de tarea realizada | Comentario de la tarea reportada |
|------------------------|-----------------------|---|---|
| 28/04/2022 23:59:42 | Mauricio Calderon | Documentación | Se realizo la primera reunion del proyecto donde se acordaron los detalles para la entrega de los primeros documentos entregables (Anexo y Plan de Proyecto) |
| 29/04/2022 0:00:38 | Mauricio Calderon | Documentación | Se determino utilizar Google Calendar como metodo de recordatorio, para evitar contratiempos en reuniones y en general para tener mejor comunicacion y trabajo en equipo. |
| 30/04/2022 14:39:47 | Yendry Badilla | Documentación | Plantilla de minutas de reunión |
| 30/04/2022 14:41:25 | Yendry Badilla | Documentación | Realización del plan de proyecto y cronograma |
| 30/04/2022 14:41:41 | Yendry Badilla | Documentación | Minuta de reunion 2 |
| 30/04/2022 16:31:22 | Gabriel Vargas | Documentación | Reunión para el plan de trabajo |
| 3/05/2022 21:05:53 | Gabriel Vargas | Documentación | Reunión para finalizar documentación |
| 3/05/2022 21:10:55 | Victor Castrillo | Documentación | Plan de trabajo - Estrategias de aprendizaje |
| 5/05/2022 18:32:58 | Yendry Badilla | Documentación | Minuta de reunion presencial |
| 5/05/2022 19:06:58 | Mauricio Calderon | Front-end (Web administrador y cliente) | Proyecto de Angular Creado |
| 5/05/2022 19:07:14 | Mauricio Calderon | Documentación | Plan de proyecto casi terminado |
| 5/05/2022 19:07:38 | Mauricio Calderon | Documentación | Se termino el modelo conceptual |
| 5/05/2022 20:49:31 | Gabriel Vargas | Documentación | Se empezó el modelo conceptual |
| 5/05/2022 20:49:44 | Gabriel Vargas | Documentación | Avance del modelo conceptual |
| 5/05/2022 20:50:19 | Yendry Badilla | Documentación | Se empezó el modelo conceptual (4/5/2022) |
| 5/05/2022 20:50:31 | Yendry Badilla | Documentación | Avance del modelo conceptual |

Aprendizaje Continuo

Estrategia de aprendizaje

Entre las herramientas que se utilizaron en el proyecto anterior están: Entity Framework, que fue utilizado para obtener los modelos y el contexto de la base de datos, Android Studio, utilizado para crear la aplicación móvil, ASP.NET para la creación del REST API y herramientas de desarrollo web como HTML5, CSS, Bootstrap y Angular.

En el proyecto anterior se utilizó PostgreSQL como base de datos en la aplicación web, por lo que algunos métodos y estrategias utilizadas para las bases de datos a implementar en este proyecto pueden servir como experiencia para no cometer los mismos errores de antes.

El trabajo incluye la implementación de funciones con herramientas que desconocemos, como MongoDB que se utilizará para guardar los comentarios y se sabe que es un sistema de base de datos NoSQL orientado a documentos de código abierto, que además no guarda los datos en tablas, sino que guarda estructuras de datos BSON. También se necesita un sistema gestor de bases de datos, que se solucionará con SQL Server para manejar un servidor que da servicio a otras aplicaciones de software. Otro ejemplo de las funciones por desarrollar incluye el uso de GPS y visualización de mapas, que es una función frecuente en el desarrollo de aplicaciones para dispositivos móviles, para mostrar las rutas de actividades creadas por los usuarios. Todo esto se subirá a la nube por medio de Azure o Amazon Web Services para almacenar los servicios requeridos por el proyecto.

En el proceso de investigación se acudirá a la documentación de cada programa disponible en la página oficial. Si no se encuentra la solución a la duda pendiente o la duda sea muy puntual, se puede acudir a videos tutoriales en YouTube y foros de consulta como Stack Overflow. Además, cuando surja una situación complicada o se gaste mucho tiempo solucionando un problema, se le consultará a algún conocido con experiencia que pueda ayudarnos. Para solventar el requerimiento de importar un Excel a un SQL server, se acudirá a la implementación del asistente de importación llamado SSIS